

134

N91-21947

The Object the Metaphor the Power and Evergreen
or The Eighth Way to Make a Hypermedia Project Fail

Bruce A. Warren
President
Warren-Forthought, inc

The Object the Metaphor the Power and Evergreen

or The Eighth Way to Make a Hypermedia Project Fail

ABSTRACT

In an effective organization neither the production process nor the worker's job stays the same for more than a few weeks. Neither can the hypermedia databases and learning tools for that job. This paper describes a patented software technique that is necessary and sufficient to keep hypermedia databases *evergreen*, or current with the manufacturing technology. The technique has proved its validity in four years of use in petrochemical plants. This technique is based on three principles, 1-the database must be object structured, i.e., all components must retain *visible individuality*, 2-the authoring process must use *no metaphors*; the author must be seeing and experiencing the multimedia data objects as he creates. And 3-the hypermedia tools must possess power in the form of *unlimited capacity*.

Tempting targets...

There are many tempting targets for hypermedia solutions. The technology appears to be here; products that might turn massive amounts of paper information into new computerized formats. However the sheer mass of static information distracts from the reality that a critical portion of it is always changing.

No job in an effective organization stays the same for more than a few weeks or months. Neither can the hypermedia databases and learning tools for that job. However, hypermedia information is harder to keep current (evergreen) than paper information because, until recently, no editing infrastructure existed.

Where are the hypermedia light tables, paste boards, whiteout, scotch tape, plate makers, printing presses, binders, word processors, etc? In the world of hypermedia this infrastructure must be entirely resident in software. This makes hypermedia authoring software so hard to develop that customers apply good tools like HyperCard to problems it can never complete. Like trying to mow Interstate 10 with a garden tractor; the first mile of freshly mowed median looks real good. Then you hit a few old tires and disappear into an overgrown culvert. Eager customers see the demo of the pretty green strip of grass, but not the 1500 miles of weeds up ahead.

And the companies with the very real and very large problems to solve are just learning to tell the difference between a garden tractor

and a John Deere 4520 Diesel with a triple gang brush hog.

...hard hat multimedia

Warren-Forthought was founded by people with the experience to know the difference. We call it 'hard hat multimedia'. Not because it makes a good marketing slogan, but because all our customers really wear hardhats.

We named our package to make it real. Our multimedia workstation is called MockKingbird®. The business plan that founded the company stressed the necessity that the tools we develop put the end-user in charge; that he be empowered to keep all his information and training Evergreen.

Neglecting the Evergreen principle is a deadly mistake, because even if the authoring system is free (like HyperCard) or the supplier is impeccable (like IBM), if you cannot make major changes to the finished product with only minutes of effort; your hypermedia application will wither away from disuse.

...change often and quickly

The competitive world forces all organizations and their business systems to change often and quickly if they are to survive. Even fast tools create multimedia products that are outdated by the time they are complete. Changes need to be made as the project proceeds and continuously as the organization evolves.

The Object the Metaphor the Power and Evergreen

or The Eighth Way to Make a Hypermedia Project Fail

Yet, with a few notable exceptions, all authoring systems and programming languages create products that are essentially unalterable in their finished state. Even minor changes open a Pandora's box of debugging, retesting, and re-validation. And consume a dispiriting amount of resources.

How many copies of 1-2-3 would Lotus have sold if it were an unchangeable generic balance sheet everyone had to use?! The true power of the personal computer lies in its ability to allow instant and infinite changes in what it does and how it delivers what it does to the user.

...Evergreen - a dichotomy

All our industrial customers accept as gospel the necessity of maintaining the equipment in their plants. The equipment must stay Evergreen or the product stops coming out the pipes. However, it is very difficult to transfer that same discipline to the mass of documentation required to run the plant right. This dichotomy evolves from the availability of a powerful substitute for good documentation - the collective memory of the folks who do the work. Lately, though, this collective memory has been making mistakes that kill people, as in the Phillips polyethylene plant explosion.

Government intervention, large numbers of old hands retiring, and good manufacturing practice are building tremendous pressure to document the plant technology in some other medium than brain cells. Everyone seems to agree that computer storage is the only hope. Paper manuals will not do the job. "No one reads them; they're always out-of-date" is the almost universal comment.

...outdated and dangerous

Whether on paper or in a computer, no matter how good a database is when released, if it cannot be maintained it will quickly become outdated and dangerous. The only practical way to maintain highly technical plant data is with the people that use the data. Only they, as a group, know when the data and the real world don't match. The technicians must have the power and permis-

sion to keep their own data correct.

Often this issue frightens the management of a plant. "...you mean those guys can change anything in here?" "What if they screw it up?" "Can I lock this disk to keep them from messing around?"

(Remember, years ago, the howls from the accountants and MIS folks when these same managers got their PC's and spreadsheets?)

But... maybe it is preferable to have a powerful and effective multimedia database with a few easily detectable errors (or even sabotage) than the 'wing and a prayer and a midnight callout' system these managers are now using. A team of technicians with a powerful and useful tool will act like a football team. They won't sabotage their teammates. (And as long as no one makes them take tests on the computer, they won't sabotage it either.)

Policy problems aside, how can it be possible for the average technician to actually maintain his own computerized information system? There are two ways to look at the answer to that question. One, teach these guys to be programmers. If you can fix multi-million dollar machinery, you probably have the mental capability to program a computer. After all it's not like these guys are entry level office workers.

..his real job

However, this is a trap that has snared many projects that attempted to 'computerize' complex documentation. We must not lose sight of the fact that, even though it can be done, becoming a programmer is not why a plant technician was hired and trained. If the computer consumes the technician's time fighting bugs and learning 'computer speak' he will not use it. His real job is to run the plant and make product. His boss never forgets that; he never forgets that.

The right answer is to seek out a database tool that becomes invisible just like any other good tool. For instance, the plant electrician does not spend hours delving into the design of his oscilloscope. He turns the knobs, uses it, and puts it back on the shelf. The mechanic doesn't get excited over his crescent wrench,

The Object the Metaphor the Power and Evergreen

or The Eighth Way to Make a Hypermedia Project Fail

he clamps it on the nut, turns it a few times and throws it back in the toolbox.

...toolness

Computer hardware and software has advanced to this state of 'toolness'. You just have to look for it and keep your computer gurus out of the way. Buy the crescent wrench, not the hacker's dream system.

The hypermedia workstation must be an integrated package, a tool, that requires no installation, or initializing, or configuration. You just plug it in, slide in a removable cartridge disk and it does all the rest. The authoring and editing tools must be packaged with interactive courses that let the technician learn the few skills he needs to be in complete control of his new tool.

Learning the new tool must take hours, not months.

The development process must be fast enough that management does not need to be persuaded to 'take a flyer' and spend a year discovering if the investment will pay off or turn into vaporware. In a month tangible useful products should appear.

...three characteristics

A computer won't become a tool in the world of a manufacturing plant floor unless it possesses three characteristics: *one-* the database must be object structured, i.e., all components must retain *visible individuality* (no stacks, no source code files, no relations, no tables, etc), *two-* the authoring process cannot use any metaphors; the author must be seeing and experiencing the multimedia data objects as he creates the hypermedia product, not some arbitrary representation of that data such as icon lists and spaghetti diagrams. And *three-* the hypermedia tools must be blessed with raw power and unlimited capacity; Lego skyscrapers are very impressive creations, but totally useless.

...visible individuality

All Mockingbird products are designed with these three principles at the core of the software design: visible individuality, no meta-

phors, and unlimited capacity. These three characteristics enable the content to stay Evergreen because the user can see it. Just like the real world. Anything hidden remains a mystery and is ignored. Visible individuality means the author and user can see each object and they are not combined into some stack or compound document or relational database. The Macintosh makes this easy by using the standard Finder. All Mockingbird objects (pictures, text, sounds, animations, etc) are placed in folders using the Finder. Nothing is hidden. Visible individuality also requires the elimination of the control program that contains the linkages and other hooks. Mockingbird software distributes the control commands and links to each of the electronic documents. No master program exists for the hypermedia database. Our patent, #4,877,404, covers this technique of embedding executable code in the individual graphics files, making each file a stand alone interactive object.

...no metaphor

The principle of *no metaphor* should be obvious, but never is. Hypermedia has never existed before. Only computers can do it. Contrast that with desktop publishing which is transplanted from a long tradition going back to Gutenberg. One could argue that the ultimate metaphor in hypermedia is virtual reality - what you see, hear and feel on the computer matches the world being documented on the computer. But the real world is not a metaphor, it is reality. If you use a metaphor like cards, icons, spaghetti diagrams, flow charts, etc. your authoring system will not be successful tackling a manufacturing process that is grounded in very real, very tangible equipment.

...power and capacity

Power and capacity limits are very hard to see in software. Subtle design decisions by software architects unfamiliar with the real world can severely limit a product. Software buyers learn to detect power and capacity limitations only with expensive failures. Hypermedia software must be limited only by

The Object the Metaphor the Power and Evergreen

or The Eighth Way to Make a Hypermedia Project Fail

disk space. (And that is almost unlimited now with networks and optical disk juke boxes). It must not slow down as the database gets larger or the number of users increases. It must work over networks with multiple users accessing a single copy of all critical multimedia objects. It must be fast; two or three seconds is all people are willing to wait to see results after a click. HyperCard, SuperCard and other similar products fail all these requirements. Mockingbird was designed over the past four years with these realities in mind; and delivers the power and capacity to handle any size project.

In practice it turns out the the capability to stay Evergreen hinges on the power and capacity issue. For instance, multiuser access to a single instance of a multimedia object is not only a critical power issue, but it enables that object to stay Evergreen. Multiple copies of the same data will kill a database faster than anything else because even good-faith efforts to update information will often result in missed locations that remain unchanged. Any old data is not only obsolete, but dangerous in an environment like a chemical plant.

Capacity must be unlimited or the users will quickly learn that it is too much work to condense, compress, and edit their new information to fit. Or they must throw out an object to fit another in. Anyone with a personal computer spends a lot of time going through this process with his hard disk. It is tough to decide what to throw away.

...back in the old days

Interestingly enough, back in the days of floppy disk based PC's this problem didn't exist. It was a different problem - how do I manage all these floppies so that I can find the one I need. Capacity additions were easy and limited only by filing space in the office - just buy a few more floppies. Think of a floppy disk as a storage object!

Evergreen requires that the hypermedia tool function like an old floppy based PC- the user sees no limits-while eliminating the search and retrieval problem created by thousands of floppies strewn everywhere.

The multimedia success equation can be stated this way:

Object Structured

+ No Metaphor

+ Huge Capacity

= Evergreen



Session 4

Research Activities at NASA

Chair: Robert T. Savely

Hypermedia Applications in a Mission Operations Environment

Tony J. Ames

Integrating Knowledge and Control into Hypermedia-Based Training Environments: Experiments with HyperCLIPS

Randall W. Hill, Jr.

Germ as a Tool for Space Station Documentation

Ken Crouse
Charles S. Harwick

